

Hung-Chen Kuo et al  
10/020160  
December 18 2001  
Birch Stewart  
Kolisch & Birch  
(703) 205-8000  
LLP

中華民國經濟部智慧財產局

INTELLECTUAL PROPERTY OFFICE  
MINISTRY OF ECONOMIC AFFAIRS  
REPUBLIC OF CHINA

Attorney Docket  
No. 3722-0115P



茲證明所附文件，係本局存檔中原申請案的副本，正確無訛，

其申請資料如下：

This is to certify that annexed is a true copy from the records of this office of the application as originally filed which is identified hereunder:

申請日：西元 2000 年 12 月 22 日  
Application Date

申請案號：089127633  
Application No.

申請人：聯發科技股份有限公司  
Applicant(s)

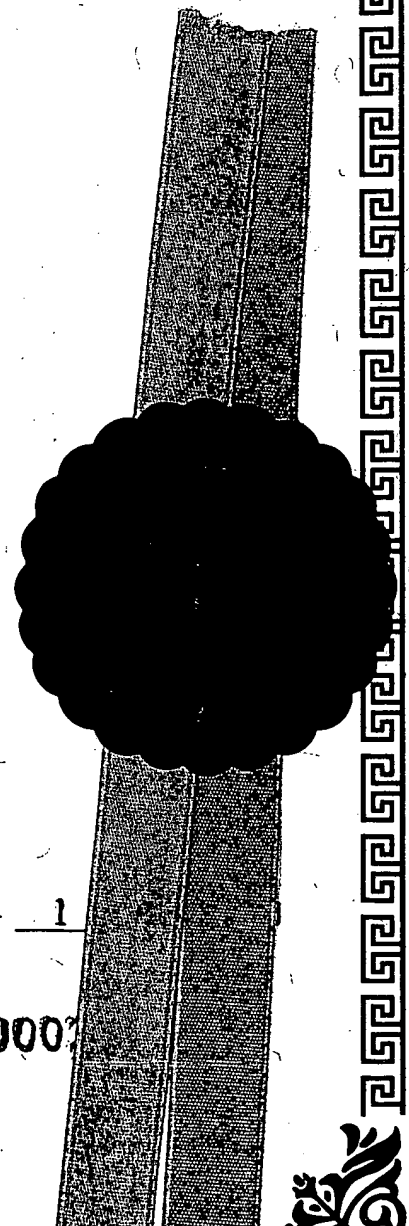
局長  
Director General

陳明邦

CERTIFIED COPY OF  
PRIORITY DOCUMENT

發文日期：西元 2002 年 1  
Issue Date

發文字號：09111000  
Serial No.



|      |  |
|------|--|
| 申請日期 |  |
| 案 號  |  |
| 類 別  |  |

A4  
C4

(以上各欄由本局填註)

# 發明專利說明書

|             |               |  |
|-------------|---------------|--|
| 一、發明<br>名稱  | 中 文           | Viterbi 解碼器的解碼電路與方法                              |
|             | 英 文           |  |
| 二、發明<br>創作人 | 姓 名           | 1 郭弘政<br>2 吳文義                                   |
|             | 國 籍           | 中華民國   |
|             | 住、居所          | 1 新竹市南大路 550 巷 50 號 3 樓<br>2 新竹縣竹北市光明十街 58 巷 5 號 |
| 三、申請人       | 姓 名<br>(名稱)   | 聯發科技股份有限公司                                       |
|             | 國 籍           | 中華民國   |
|             | 住、居所<br>(事務所) | 新竹科學工業園區新竹縣創新一路 13 號 1 樓                         |
|             | 代 表 人<br>姓 名  | 蔡明介  |

裝

訂

線

## 四、中文發明摘要 (發明之名稱: Viterbi 解碼器的解碼電路與方法)

一種 Viterbi 解碼器的解碼電路與方法，此 Viterbi 解碼器的解碼電路包括一個 Branch Metric 單元、一個加-比較-選擇單元與一個路徑記憶體單元、此路徑記憶體單元包括一個資料流向控制器、一個追溯寫入暫存器陣列、一個等待暫存器陣列及一個解碼暫存器陣列。利用游程長度限制碼有效解決 Viterbi 解碼器的格狀圖經縱向整理後複雜的格狀圖，而且暫存器陣列在不同的時間可兼具其他動作，所以不需要數量很多的暫存器來處理資料，同樣地可以達到高速 Viterbi 解碼器之解碼的目的。

## 英文發明摘要 (發明之名稱: )



(請先閱讀背面之注意事項再填寫本頁各欄)

裝

訂

線

## 五、發明說明 ( / )

本發明是有關於一種光碟的 PRML(Partial Response Maximum Likelihood)系統中，Viterbi 解碼器的解碼電路與方法，且特別是有關於一種光碟的 PRML 系統中，利用游程長度限制碼(Run Length Limited Code, RLL Code)有效解決 Viterbi 解碼器的格狀圖經縱向整理後複雜的格狀圖，而且不需要數量很多的暫存器來處理資料的 Viterbi 解碼器的解碼電路與方法。

在光碟如 DVD(digital versatile disc)的 PRML 系統中，對於一個具有記憶體的傳輸頻道(transmission channel)可以用格狀圖(trellis diagram)來描述其特性。例如 DVD 頻道的輸入信號 EFMP 是二進位信號，而頻道記憶體長度(channel memory length)為 2，因此格狀圖上有 4 個狀態(state)，每個狀態有兩支分支(branch)。在每一個狀態中包含了 8 條分支，表示此傳輸頻道在每次的單位時間中傳出的信號，會是這八個可能信號中的其中一個，至於這八種信號的值，如第 1A 圖繪示頻道模型示意圖所示，由此頻道模型(channel model)可知，並將下列運算式代入  $1/-1$  即可得到：

(“目前輸入”+  $k_1$  \* ”在第一個記憶體之先前輸入”+  $k_2$  \* ”在第二個記憶體之先前輸入”)

在第 1A 圖中， $k_1$  與  $k_2$  實施上反應出此頻道的特性，至於  $k_1$  與  $k_2$  該是何值可以預先給定，然後用整型濾波器(shaping filter)來使總體頻道的特性趨近於該值(整型濾波器即是部分響應等化器(partial response equalizer))。

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

## 五、發明說明(2)

另一種則是不去改變頻道的特性，而是直接估算  $k_1$  與  $k_2$  的值(即信號位階估算(signal level estimation))。不論採取何種方式，總之在第 1B 圖繪示格狀圖中各分支上的對應值，將因  $k_1$  與  $k_2$  的確定之後而可以確定。

第 2 圖繪示 Viterbi 解碼器方塊圖。在第 2 圖中，Viterbi 解碼器 200 是包括 Branch Metric 單元 202、ACS (Add-Compare-Select) 204 與路徑記憶體單元 (Path Memory Unit, PMU) 206。Branch Metric 單元 202 接收前級轉換器(未繪示)所輸出的數位信號與參考位階產生單元(未繪示)所輸出的參考位階，並計算一 branch metric 及輸出此 branch metric 至 ACS 204。ACS 204 根據 branch metric 做加法、比較與選擇等運算以得以一 path metric，並且將此 path metric 輸出至路徑記憶體單元 206。路徑記憶體單元 206 接收 ACS 204 所輸出的 path metric，進行收斂、合併與解碼等工作以得到解碼信號，並且將此解碼信號輸出至後級的解調器(未繪示)。以下將對 Viterbi 解碼器 200 做更詳細的說明。

第 3 圖繪示 Viterbi 解碼器的解碼演繹。在第 3 圖中，假如收到三個連續接收信號 R1、R2、R3(received signals R1、R2、R3)，要找到對應之最有可能的信號，首先針對格狀圖中每一條分支，Viterbi 解碼器會算出 branch metric(由第 2 圖的 Branch Metric 單元 202 執行)。計算的方式通常為信號差的平方值或是信號差的絕對值，例如  $BM_{(m)-(n)} = [R1 - (-1.8)]^2$  或  $|R1 - (-1.8)|$ 。而所謂某一條路

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

線

## 五、發明說明 ( 3 )

徑的 path metric 就是將此路徑所含的所有分支所對應的 branch metric 加起來，例如第 3 圖中粗黑線由  $State_{00\ at\ 1T}$  經過  $State_{00\ at\ 2T}$  再經過  $State_{00\ at\ 3T}$  最後到達  $State_{00\ at\ 4T}$  的這條路徑，而 path metric 就是 " $BM_{00-00} + BM_{00-01} + BM_{01-11}$ "。

當然，在一般情況下路徑不會如此短，如上述的路徑在  $State_{00\ at\ 1T}$  之前應該就已經走過一段路徑，所以實際上這條路徑正確的 path metric 應該是 "此路徑累計到  $State_{00\ at\ 1T}$  時的 path metric" + " $BM_{00-00} + BM_{00-01} + BM_{01-11}$ "。再觀察格狀圖可知，其實各路徑不斷的相交於各狀態上，而任意路徑在相交之後的變化其實都是相同的，因此 Viterbi 發現只要在各狀態選擇此處的最佳路徑繼續延伸即可，保證最後仍然可算出最佳路徑。但是運算複雜度卻可因此而大幅降低。這種 "各狀態選擇此處的最佳路徑繼續延伸" 的動作稱為存活路徑選擇 (survivor path selection)，而選出來的路徑就是存活路徑。

在第 3 圖中，描述在  $State_{00\ at\ 2T}$  的存活路徑是如何由來自  $State_{00\ at\ 1T}$  的存活路徑與來自  $State_{10\ at\ 1T}$  的存活路徑挑選出來，參考第 3 圖，在  $State_{00\ at\ 2T}$  的 survivor metric  $SM_{00\ at\ 2T}$  是根據下列運算而得到：

$$SM_{00\ at\ 2T} = \min\{[SM_{00\ at\ 1T} + BM_{00-00}]; [SM_{10\ at\ 1T} + BM_{10-00}]\}$$

這個運算式就是 ACS 單元。由於 ACS 單元內存在著迴授回路 (feedback loop) (由 survivor metric 每個狀態都會再次重算所造成的)，使得高速的 Viterbi 解碼器不易實現，而 ACS 因此成為速度的瓶頸。

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

線

## 五、發明說明(4)

根據 Viterbi 解碼器的演繹，在每次 ACS 的動作之後，必定可以取得最佳路徑，所以只要不斷的記錄各存活路徑，等到整筆資料解碼結束後，再轉移出最佳存活路徑的記錄就可以。但如此有兩個缺點，首先是解碼隱藏路徑會太長，其次，如果路徑太長，則用來記憶的硬體數量會很大(此硬體稱為路徑記憶體單元)。

如第 4 圖繪示在格狀圖中使用 Viterbi 解碼器的演繹解碼之收斂示意圖所示，Viterbi 演繹有個特性就是在格狀延展過 4~6 倍頻道記憶(channel memory)的長度之後，各個存活路徑有 99%的機會是與前面的路徑會合併(merge)在一起。因此路徑記憶體單元只需保存最近 4~6 倍頻道記憶(channel memory)的資料以供撿選即可，過去的資料可在逐步解碼之後便可丟棄。因此，只需要有限的資源就可以完成路徑記憶體單元的工作，而常見的方法有混合交換(Shuffle-Exchange)與回溯(Trace-Back)兩種。

第 5 圖繪示格狀圖與混合交換方法之方塊圖。在第 5 圖中，混合交換的動作原理非常直覺，就是對於格狀圖上的每一個狀態配備一套記憶體(通常是暫存器陣列(register array))，如狀態"00"對應於暫存器陣列 502，狀態"01"對應於暫存器陣列 504，狀態"10"對應於暫存器陣列 506，狀態"11"對應於暫存器陣列 508，這些暫存器陣列用來記錄此狀態的存活路徑。

也就是說，隨著時間的前進，在任一特定時刻上各

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

線

## 五、發明說明(5)

個狀態所記錄的存活路徑會不斷地由前一時刻的各狀態所記錄的存活路徑配上新判斷的判斷位元(如第 5 圖所示 -1.8、-0.8、0.2、1.2、-1.2、-0.2、0.8、1.8 等數值)來更迭。如第 5 圖所示，當信號接收數量經過 4~6 倍頻道記憶體左右的長度之後，便假設這所有的路徑均以收斂到同一點，於是可將收斂得到的數值當成解碼所成爲的信號送出。因此，實際上每個暫存器陣列的長度只須 4~6 倍頻道記憶體左右。

如第 6 圖繪示格狀圖及回溯方法方塊圖所示，回溯的動作原理是將各狀態(每一個時間點)各狀態上的判斷位元(decision bit)存在"規則的記憶體"中(例如 RAM)，即各個狀態各自規律的存放其判斷位元(例如狀態"00"存放在 RAM 第 1 列，狀態"01"存放在 RAM 第 2 列，以下以此類推)，各狀態間沒有互動，經過 4~6 倍頻道記憶體的長度之後，如第 4 圖所示，有合併點(即 root)出現。於是可由任意狀態起，運用存下的判斷位元倒推算出前一個狀態，然後依序一路回溯可找到合併點。

然而在實施運用中，不可能每一次解一個合併點都動用如此一次的回溯，而是利用分批作業的方法，將資料安排成"解到合併點之後，然後一路解出一筆資料"的組態。其可用四個動作來描述：V(寫入(write-in))、T(追溯(trace))、I(等待(idle))、D(解碼 decode))。

如第 7 圖繪示回溯方法的四個基本動作之時間空間關係圖所示，首先以垂直方向來看，記憶體 M1 在 T1 時，

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線



## 五、發明說明(6)

規律地將各狀態的判斷位元寫入(V)。接著，在 T2 由某個狀態開始回溯到合併點，找到合併點便交給記憶體 M4 (在格狀上記錄相鄰區塊，然後在區塊 M4 下次就把合併點以後的判斷位元解出來(D))。在 T3 時，記憶體 M1 的資料暫時等待(I)，因為離上次寫入(V)時間還未久，因此要等格狀發生合併的狀況，然後在 T4 時間，記憶體 M1 內的資料終於算是合併。此時，記憶體 M2 所產生的合併點給記憶體 M1 以開始進行解碼(D)。

再以水平方向來看，在某個時間點，各記憶體區塊分別進行 V、D、I、T 四個動作。回溯方法的優點在於其規律的動作，同一級(stage)的各個狀況是各自存各自的，因此佈局(layout)的困難度降低。缺點是通常以 RAM 來實現，所以用在 DVD Player 還可以，但用在 DVD ROM 則嫌太慢，如果用暫存器來增加速度，則因回溯方法是分成四個區塊同時處理資料，要付出相當高的硬體代價，並不適合較貴的記憶體。

因此本發明係提供一種 Viterbi 解碼電路與方法，可以解決 Viterbi 格狀圖經縱向整理後複雜的格狀圖，而且不需要數量很多的暫存器來處理資料同樣可以達到高速 Viterbi 解碼的目的。

本發明係提供一種 Viterbi 解碼器的解碼方法，此 Viterbi 解碼器具有 Branch Metric 單元、加-比較-選擇單元與路徑記憶體單元，此方法的步驟包括：首先，對配合 Viterbi 解碼器的格狀圖進行縱向整理，以得到一縱向

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

線

## 五、發明說明( 7 )

整理格狀圖。接著，根據游程長度限制碼運算此縱向整理格狀圖，以得到一游程長度限制格狀圖。其次，根據此游程長度限制格狀圖令 Branch Metric 單元計算一目前輸入 branch metric 值。再者，由加-比較-選擇單元計算此目前輸入 branch metric 值、一下次輸入 branch metric 值與一目前狀態值，以得到一下次狀態值。以及，以路徑記憶體單元記錄加-比較-選擇單元所計算的結果，以成爲一存活路徑，將記錄完成的存活路徑進行解碼。

本發明係提供另一種 Viterbi 解碼器的解碼方法，此 Viterbi 解碼器具有 Branch Metric 單元、加-比較-選擇單元與路徑記憶體單元，此路徑記憶體單元包括解碼暫存器陣列、追溯寫入暫存器陣列、等待暫存器陣列及加-比較-選擇單元，此方法的步驟包括：首先，建立配合 Viterbi 解碼器的格狀圖。接著，對配合此 Viterbi 解碼器的格狀圖進行縱向整理，以得到一縱向整理格狀圖。然後，根據一游程長度限制碼運算此縱向整理格狀圖，以得到一游程長度限制格狀圖。其次，根據此游程長度限制格狀圖令 Branch Metric 單元計算一 branch metric 值。再者，由加-比較-選擇單元計算一目前輸入 branch metric 值、一下次輸入 branch metric 值與一目前狀態值，以得到一下次狀態值。接著，由加-比較-選擇單元僅根據此下次狀態值設定一目前輸出判斷位元與一下次輸出判斷位元。再者，根據此目前輸出判斷位元與此下次輸出判斷位元記錄上次的狀態值與目前的狀態值

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

## 五、發明說明( 8 )

上之一判斷位元。其次，判斷目前的狀態值與判斷位元的關係，將判斷位元寫入適當的狀態值。然後，由加-比較-選擇單元送出判斷位元至追溯寫入暫存器陣列，並追溯得到一合併值。接著，儲存在解碼暫存器陣列的判斷位元到預定數量時，解碼暫存器陣列變為等待暫存器陣列。以及，根據合併值解碼原先的等待暫存器陣列所儲存之判斷位元。

本發明提出一種 Viterbi 解碼器的解碼電路，此解碼電路具有 Branch Metric 單元、加-比較-選擇單元與路徑記憶體單元，Branch Metric 單元用以計算一 branch metric 值，並輸出此 branch metric 值至加-比較-選擇單元。加-比較-選擇單元根據此 branch metric 值以運算一狀態值的最佳路徑，並輸出一判斷位元。路徑記憶體單元包括：一個資料流向控制器用以接收加-比較-選擇單元所輸出的判斷位元，並決定判斷位元的輸出流向。一個追溯寫入暫存器陣列接收資料流向控制器所輸出的判斷位元，並追溯得到一合併值。一個等待暫存器陣列對追溯寫入暫存器陣列所追溯得到的合併值以進行解碼。以及，一個解碼暫存器陣列根據已追溯得到的合併值，將儲存在解碼暫存器陣列的判斷位元進行解碼。

因此本發明係提供一種 Viterbi 解碼器的解碼電路與方法，利用游程長度限制碼有效解決 Viterbi 解碼器的格狀圖經縱向整理後複雜的格狀圖，使 ACS 可以平行進行加法與比較運算，而且暫存器陣列在不同的時間可兼

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

## 五、發明說明(9)

具其他動作，如此，ACS 與 PMU 不需要數量很多的暫存器來處理資料，同樣地可以達到高速 Viterbi 解碼器之解碼的目的。

為讓本發明之上述目的、特徵、和優點能更明顯易懂，下文特舉較佳實施例，並配合所附圖式，作詳細說明如下：

圖式之簡單說明：

第 1A 圖繪示頻道模型示意圖；

第 1B 圖繪示格狀圖；

第 2 圖繪示 Viterbi 解碼器方塊圖；

第 3 圖繪示 Viterbi 解碼器的解碼演繹；

第 4 圖繪示在格狀圖中使用 Viterbi 解碼器的演繹解碼之收斂示意圖；

第 5 圖繪示格狀圖與混合交換方法之方塊圖；

第 6 圖繪示格狀圖及回溯方法方塊圖；

第 7 圖繪示回溯方法的四個基本動作之時間空間關係圖；

第 8 圖繪示本發明 Viterbi 解碼器之方法的流程圖；

第 9A 圖繪示原始格狀圖；

第 9B 圖繪示將原始格狀圖做橫向整理；

第 9C 圖繪示將原始格狀圖做縱向整理；

第 10 圖繪示二進位輸入之頻道記憶體為 3 的格狀圖；

第 11 圖繪示有 RLL(2, 10)碼的限制下之格狀圖；

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

## 五、發明說明(10)

第 12 圖繪示將第 9 圖做縱向整理的結果；

第 13 圖繪示將第 9 圖做兩次縱向整理的結果；

第 14 圖繪示以第一種回溯方法解格狀圖之關係對應圖與狀態擷取電路圖；

第 15 圖繪示以第二種回溯方法解格狀圖之關係對應圖與狀態擷取電路圖；

第 16 圖繪示以第三種回溯方法解格狀圖之關係對應圖與狀態擷取電路圖；

第 17A 圖繪示回溯方法使用暫存器陣列的電路方塊圖；

第 17B 圖繪示回溯方法的資料流向；

第 17C 圖繪示回溯方法的另一種資料流向；以及

第 17D 圖繪示在資料串加入虛資料延遲之示意圖。

標號說明：

200：Viterbi 解碼器(Viterbi decoder)

202：Branch Metric 單元(Branch Metric unit)

204：ACS

206：路徑記憶體單元(path memory unit)

502，504，506，508，1506，1508，1510：暫存器陣列(register array)

1400，1500，1600，1712，1716，1718，1722：狀態擷取電路(state retrieve circuit)

1402，1502，1602：判斷電路(decision circuit)

1404，1504，1604：狀態(state)

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

## 五、發明說明 ( // )

1406, 1410, 1506, 1510, 1606, 1610 : 多工器  
(multiplexer)

1408, 1508, 1608 : 記憶體區塊(memory block)

1700 : 回溯電路(trace-back circuit)

1702 : 加-比較-選擇單元(add-compare-select unit)

1704 : 資料流向控制器(data string controller)

1714, 1720 : 延遲暫存器(delay register)

### 實施例

第 8 圖繪示本發明 Viterbi 解碼器之方法的流程圖。在第 8 圖中，首先建立配合 Viterbi 解碼器的格狀圖(S70)(如第 3 圖所示)。為了解決 ACS 的瓶頸乃對 Viterbi 解碼器的格狀圖做重整，而重整的方法包括橫向整理與縱向整理兩種方式。以第 9A 圖繪示原始格狀圖且  $n=2$  為例，在第 9B 圖繪示將原始格狀圖做橫向整理，所謂橫向整理就是將原先  $n$  個狀態的格狀合併為一個狀態，雖然 ACS 的瓶頸同樣存在，但卻將瓶頸的時間由  $1T$  拉長為  $nT$ ，整個 Viterbi 解碼器的速度限制也因此獲得舒解。

第 9C 圖繪示將原始格狀圖做縱向整理。在第 9C 圖中，對配合 Viterbi 解碼器的格狀圖進行縱向整理(S72)。因為由縱向整理得到的縱向整理格狀圖，其狀態的數目會 2 的指數增加，使縱向整理格狀圖變的複雜，故使用游程長度限制碼來限制某些狀態不能出現，以運算縱向整理格狀圖而得到一游程長度限制格狀圖(S74)。

## 五、發明說明 ( /2)

如第 10 圖繪示二進位輸入之頻道記憶體為 3 的格狀圖所示，此頻道與一般常見的 PRML 頻道頗為近似。以第 11 圖繪示有 RLL(2, 10)碼的限制下之格狀圖為例，將第 10 圖的格狀圖做 RLL(2, 10)碼的限制，可以發現第 11 圖的格狀圖已經簡化很多。如第 12 圖繪示將第 11 圖做縱向整理的結果所示，將第 11 圖的格狀圖做縱向整理與 RLL(2, 10)碼的限制，可以發現每一級狀態數目僅增加為 2。如第 13 圖繪示將第 11 圖做兩次縱向整理的結果所示，將第 11 圖的格狀圖做兩次縱向整理與 RLL(2, 10)碼的限制，可以發現格狀圖並未變的太複雜，但對 ACS 的瓶頸已經大幅放寬。

接著，針對格狀圖中每一條分支，Viterbi 解碼器會計算出 branch metric 值(由第 2 圖的 Branch Metric 單元 202 執行)(如第 8 圖步驟 S76)。其次，縱向整理是將 Viterbi 解碼器視為一部有限狀態機器(Finite State Machine, FSM)，由有限狀態機器的特性可知：

目前輸出=函數(目前輸入與目前狀態)

下次狀態=函數(目前輸入與目前狀態)

若將狀態重新定義，把新的目前狀態看成”原本的目前狀態+原本的目前輸入”，即把原本的輸入吸收到狀態內，則變成：

目前輸出=僅是下次狀態的函數

下次狀態=函數(目前輸入與目前狀態)

由第 9C 圖可知，由於格狀圖變成具有目前輸出=僅是下

(請先閱讀背面之注意事項再填寫本頁)

裝 · 訂 · 線

## 五、發明說明 (13)

次狀態的函數的特性，因此在找尋存活路徑時(此時候選路徑(candidate path)交會在同一個下次狀態上，所以他們的輸出信號將會一樣)，ACS 的瓶頸將會消失。因為，此時連接在此狀態上的所有分支都有相同的信號值(因為目前輸出=僅是下次狀態的函數，而下次狀態相同)，所以在計算新的 path metrics(=舊的 path metrics + branch metric)的同時，就可比較舊的 path metrics 來決定新的 path metric 之大小關係。

這樣的做法可以繼續推演，以達到任何系統需要的速度，例如將狀態重新定義再擴張一倍，把新的目前狀態看成“現在的目前狀態+現在的目前輸入+下次的輸入”，即把現在的輸入與下次的輸入吸收到狀態內，變成：

(目前輸出；下次輸出)=僅是下次狀態的函數(如第 8 圖步驟 S78)

下次狀態=函數(下次輸入、目前輸入與目前狀態)(如第 8 圖步驟 S80)

與前述是同樣的道理，雖然此狀況下的輸出變成兩個，但確定所有交會的路徑在這個狀態的信號值仍然是一樣的。上述的做法是藉由狀態的再擴張達到更快速的效果，使 ACS 的瓶頸更加鬆弛。

本實施例提出回溯的方法，是針對在 Viterbi 演繹用游程長度限制碼來找尋格狀圖的最佳路徑，提供所需的路徑記憶體管理機制。如上所述，回溯方法的工作原

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線



## 五、發明說明 (14)

理就是先規律的記錄各狀態的判斷位元，然後再藉著這些記錄回溯合併點。事實上，只有回溯的過程中，會造成混淆的地方才需要記錄(參考第 8 圖之步驟 S82)。

第 14 圖繪示以第一種回溯方法解格狀圖之關係對應圖與狀態擷取電路圖。在第 14 圖中，經過游程長度限制碼的限制，隨著一些狀態的消失，有些分支也消失了。存留下來的狀態其相互連接關係變得十分明確。例如，狀態”1000”必定來自狀態”1100”，這樣的連線根本不需要記錄在回溯方法時，也能明確的逆推得到。

以第 14 圖為例，在記錄判斷位元時(寫入(V)的動作)，只須使用兩套記憶體而不是八套記憶體。在讀回記錄時(追溯(T)、解碼(D))，狀態擷取電路 1200 中之判斷電路 1402 取狀態 1404 的前三個位元之狀態值，判斷目前的狀態 1404 之狀態值與所記錄的判斷位元之關係(即判斷狀態值是否為合法的狀態值)(參考第 8 圖之步驟 S84)。當狀態 1404 的前三位元為”000”或”111”時，表示要判斷，並藉由先前記錄下來的判斷位元繼續回溯(參考第 8 圖之步驟 S86)，如狀態 1404 的第一個位元為”1”，則多工器 1406 選擇記憶體區塊 1408 上面一排的資料輸出至多工器 1410，並由判斷電路 1402 控制多工器 1410 輸出資料至狀態 1404；如狀態 1404 的第一個位元為”0”，則多工器 1406 選擇記憶體區塊 1408 下面一排的資料輸出至多工器 1410，並由判斷電路 1402 控制多工器 1410 輸出資料至狀態 1404。當狀態 1404 的前三位元既不

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

## 五、發明說明(15)

是"000"也不是"111"時，只需要延伸狀態 1404 的第一個位元即可，如狀態 1404 的第一個位元之狀態值輸出至多工器 1410，並由判斷電路 1402 控制多工器 1410 輸出資料至狀態 1404 以產生新的狀態 1404 之狀態值。

第 15 圖繪示以第二種回溯方法解格狀圖之關係對應圖與狀態擷取電路圖，第 16 圖繪示以第三種回溯方法解格狀圖之關係對應圖與狀態擷取電路圖。第 15 圖與第 16 圖分別以更複雜的格狀圖，以下簡單地描述第 15 圖與第 16 圖工作原理。

在第 15 圖中，狀態擷取電路 1500 中之判斷電路 1502 判斷狀態 1504 的前二個位元之狀態值是否合法，當狀態 1504 的前二位元為"00"或"11"時表示合法，如狀態 1504 的第一個位元與第三個位元決定多工器 1506 選擇記憶體區塊 1508 其中一排的資料輸出至多工器 1510，並由判斷電路 1502 控制多工器 1510 輸出資料至狀態 1504。當狀態 1504 的前二位元既不是"00"也不是"11"時表示不合法，則延伸狀態 1504 的第一個位元之狀態值兩次並輸出至多工器 1510，並由判斷電路 1502 控制多工器 1510 輸出資料至狀態 1504 以產生新的狀態 1504 之狀態值。

同理，在第 16 圖中，狀態擷取電路 1600 中之判斷電路 1602 判斷狀態 1604 的前三個位元之狀態值是否合法，當狀態 1604 的前三位元為"000"或"111"時表示合法，如狀態 1604 的第一個位元、第四個位元與第五個

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

## 五、發明說明 (16)

位元決定多工器 1606 選擇記憶體區塊 1608 其中一排的資料輸出至多工器 1610，並由判斷電路 1602 控制多工器 1610 輸出資料至狀態 1604。當狀態 1604 的前三位元既不是"000"也不是"111"時表示不合法，則延伸狀態 1604 的第一個位元之狀態值並輸出至多工器 1610，並由判斷電路 1602 控制多工器 1610 輸出資料至狀態 1604 以產生新的狀態 1604 之狀態值。

第 17A 圖繪示回溯方法使用暫存器陣列的電路方塊圖。在第 17A 圖中，回溯電路 1700 包括加-比較-選擇單元 1702，用以運算各個狀態的最佳路徑並輸出一判斷位元。資料流向控制器 1704 用以接收加-比較-選擇單元 1702 所輸出的判斷位元，並決定判斷位元的輸出流向。暫存器陣列 1706 用以接收資料流向控制器 1704 所輸出的每一筆判斷位元，且從暫存器陣列 1706 移出一筆先前的判斷位元，並做追溯的動作以得到一合併值。暫存器陣列 1708 或暫存器陣列 1710 等待暫存器陣列 1706 所追溯得到的合併值以進行解碼，以及，並移出解碼後的判斷位元以作為資料輸出。

狀態擷取電路 1712 暫存器陣列 1706 做雙向資料傳輸，並送出資料至延遲暫存器 1714，延遲暫存器 1714 接收狀態擷取電路 1712 所送出的資料，並延遲一段時間後輸出資料至狀態擷取電路 1716，狀態擷取電路 1716 接收延遲暫存器 1714 所送出的資料，並與暫存器陣列 1710 做雙向資料傳輸。狀態擷取電路 1718 與暫存器陣

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

## 五、發明說明(17)

列 1706 做雙向資料傳輸，並送出資料至延遲暫存器 1720，延遲暫存器 1720 接收狀態擷取電路 1718 所送出的資料，並延遲一段時間後輸出資料至狀態擷取電路 1722，狀態擷取電路 1722 接收延遲暫存器 1720 所送出的資料，並與暫存器陣列 1708 做雙向資料傳輸。

第 17B 圖繪示回溯方法的資料流向。在第 17B 圖中，參考第 17A 圖，當資料流向控制器 1704 控制資料的流向是往左邊時，則暫存器陣列 1710 為等待(I)狀況，暫存器陣列 1708 為解碼(D)狀況，暫存器陣列 1708 在解碼時並將解碼好的資料往左邊一筆一筆地移出(參考第 8 圖之步驟 S94)。暫存器陣列 1706 從左邊寫入(V)一筆一筆的判斷位元資料，並且被暫存器陣列 1706 移出的資料往左邊移入暫存器陣列 1708，當暫存器陣列 1708 填滿移入的資料時，暫存器陣列 1708 由解碼(D)狀況變為等待(I)狀況(參考第 8 圖之步驟 S92)。此時，暫存器陣列 1706 亦做追溯(T)的動作，以得到合併點供暫存器陣列 1710 下一次解碼用(參考第 8 圖之步驟 S90)。

同理，第 17C 圖繪示回溯方法的另一種資料流向。在第 17C 圖中，參考第 17A 圖，當資料流向控制器 1704 控制資料的流向是往右邊時，則暫存器陣列 1708 為等待(I)狀況，暫存器陣列 1710 為解碼(D)狀況，暫存器陣列 1710 在解碼時並將解碼好的資料往右邊一筆一筆地移出(參考第 8 圖之步驟 S94)。暫存器陣列 1706 從右邊寫入(V)一筆一筆的判斷位元資料，並且被暫存器陣列

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線

## 五、發明說明(18)

1706 移出的資料往右邊移入暫存器陣列 1710，當暫存器陣列 1708 填滿移入的資料時，暫存器陣列 1708 由解碼(D)狀況變為等待(I)狀況(參考第 8 圖之步驟 S92)。此時，暫存器陣列 1706 亦做追溯(T)的動作，以得到合併點供暫存器陣列 1708 下一次解碼用(參考第 8 圖之步驟 S90)。

第 17D 圖繪示在資料串加入虛資料延遲之示意圖。在第 17A 圖中，例如，暫存器陣列 1706 藉由狀態擷取電路 1712、延遲暫存器 1714 與狀態擷取電路 1716 傳輸資料給暫存器陣列 1710，會產生資料傳輸延遲現象，必須在暫存器中放入數個位元的虛資料(dummy)，避免解碼得到的資料因傳輸延遲而產生資料錯誤。

因此，本發明的優點係提供一種 Viterbi 解碼器的解碼電路與方法，利用游程長度限制碼有效解決 Viterbi 解碼器的格狀圖經縱向整理後複雜的格狀圖，使 ACS 可以平行進行加法與比較運算，而且暫存器陣列在不同的時間可兼具其他動作，如此，ACS 與 PMU 不需要數量很多的暫存器來處理資料，同樣地可以達到高速 Viterbi 解碼器之解碼的目的。

綜上所述，雖然本發明已以較佳實施例揭露如上，然其並非用以限定本發明，任何熟習此技藝者，在不脫離本發明之精神和範圍內，當可作各種之更動與潤飾，因此本發明之保護範圍當視後附之申請專利範圍所界定者為準。

## 六、申請專利範圍

1.一種 Viterbi 解碼器的解碼方法，該 Viterbi 解碼器具有一 Branch Metric 單元、一加-比較-選擇單元與一路徑記憶體單元，該方法的步驟包括：

對配合該 Viterbi 解碼器之一格狀圖進行縱向整理，以得到一縱向整理格狀圖；

根據一游程長度限制碼運算該縱向整理格狀圖，以得到一游程長度限制格狀圖；

根據該游程長度限制格狀圖令該 Branch Metric 單元計算一目前輸入 branch metric 值；

由該加-比較-選擇單元計算該目前輸入 branch metric 值、一下次輸入 branch metric 值與一目前狀態，以得到一下次狀態值；以及

以該路徑記憶體單元記錄該加-比較-選擇單元所計算的結果，以得到一存活路徑，將記錄完成之該存活路徑進行解碼。

2.如申請專利範圍第 1 項所述之 Viterbi 解碼器的解碼方法，其中在進行縱向整理之前先建立配合該 Viterbi 解碼器之該格狀圖。

3.如申請專利範圍第 2 項所述之 Viterbi 解碼器的解碼方法，其中將配合該 Viterbi 解碼器之該格狀圖做縱向整理係採多次縱向整理方式。

4.如申請專利範圍第 3 項所述之 Viterbi 解碼器的解碼方法，其中由該加-比較-選擇單元僅根據該下次狀態值設定一目前輸出判斷位元與一下次輸出判斷位元。

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

線

## 六、申請專利範圍

5.如申請專利範圍第 4 項所述之 Viterbi 解碼器的解碼方法，其中根據該游程長度限制格狀圖所計算出該 branch metric 值輸入至該加-比較-選擇單元，以成為該目前輸入 branch metric 值及該下次輸入 branch metric 值二者其中之一。

6.如申請專利範圍第 5 項所述之 Viterbi 解碼器的解碼方法，其中該路徑記憶體單元記錄該存活路徑的方法係使用一混合交換法與一回溯法二者其中之一。

7.如申請專利範圍第 6 項所述之 Viterbi 解碼器的解碼方法，其中該混合交換法是對於配合該 Viterbi 解碼器之該格狀圖的每一個狀態配備一套記憶體，以用來記錄到達此狀態值之該存活路徑。

8.如申請專利範圍第 6 項所述之 Viterbi 解碼器的解碼方法，其中該回溯法更包括下列之步驟：

根據該目前輸出判斷位元與該下次輸出判斷位元記錄上次之一狀態值與目前之該狀態值之一判斷位元；

判斷目前之該狀態值與該判斷位元之關係；

將該判斷位元寫入適當之該狀態值；

當不是記錄該判斷位元之該狀態時，則寫入與該狀態的第一個位元相同之一延伸位元至該狀態；

送出該判斷位元至一追溯寫入暫存器陣列，並追溯得到一合併值；

儲存在一解碼暫存器陣列之該判斷位元到預定數量時，該解碼暫存器陣列變為一等待暫存器陣列；以及

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

線

## 六、申請專利範圍

根據該合併值解碼原先之該等待暫存器陣列所儲存之該判斷位元。

9.一種 Viterbi 解碼器的解碼方法，該 Viterbi 解碼器具有一 Branch Metric 單元、一加-比較-選擇單元與一路徑記憶體單元，該路徑記憶體單元包括一解碼暫存器陣列、一追溯寫入暫存器陣列及一等待暫存器陣列，該方法的步驟包括：

建立配合該 Viterbi 解碼器之一格狀圖；

將配合該 Viterbi 之該格狀圖進行縱向整理，以得到一縱向整理格狀圖；

根據一游程長度限制碼運算該縱向整理格狀圖，以得到一游程長度限制格狀圖；

根據該游程長度限制格狀圖令該 Branch Metric 單元計算一 branch metric 值；

由該加-比較-選擇單元計算一目前輸入 branch metric 值、一下次輸入 branch metric 值與一目前狀態值，以得到一下次狀態值；

由該加-比較-選擇單元僅根據該下次狀態值設定一目前輸出判斷位元與一下次輸出判斷位元；

根據該目前輸出判斷位元與該下次輸出判斷位元記錄上次之一狀態值與目前之該狀態值上之一判斷位元；

判斷目前之該狀態與該判斷位元之關係；

將該判斷位元寫入適當之該狀態值；

由該加-比較-選擇單元送出該判斷位元至該追溯寫

(請先閱讀背面之注意事項再填寫本頁)

裝  
訂  
線



## 六、申請專利範圍

入暫存器陣列，並追溯得到一合併值；

儲存在該解碼暫存器陣列之該判斷位元到預定數量時，該解碼暫存器陣列變為該等待暫存器陣列；以及

根據該合併值解碼原先之該等待暫存器陣列所儲存之該判斷位元。

10.如申請專利範圍第 9 項所述之 Viterbi 解碼器的解碼方法，其中將配合該 Viterbi 解碼器之該格狀圖做縱向整理，係採用多次縱向整理。

11.如申請專利範圍第 10 項所述之 Viterbi 解碼器的解碼方法，其中根據該游程長度限制格狀圖所計算出該 branch metric 值輸入至該加-比較-選擇單元，以成為該目前輸入 branch metric 值及該下次輸入 branch metric 值二者其中之一。

12.一種 Viterbi 解碼器的解碼電路，該解碼電路具有一 Branch Metric 單元、一加-比較-選擇單元與一路徑記憶體單元，該 Branch Metric 單元用以計算一 branch metric 值，並輸出該 branch metric 值至該加-比較-選擇單元，該加-比較-選擇單元根據該 branch metric 值以運算一狀態值的最佳路徑，並輸出一判斷位元，該路徑記憶體單元包括：

一資料流向控制器，用以接收該加-比較-選擇單元所輸出之該判斷位元，並決定該判斷位元的輸出流向；

一追溯寫入暫存器陣列，接收該資料流向控制器所輸出之該判斷位元，並追溯得到一合併值；

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

線

## 六、申請專利範圍

一等待暫存器陣列，對該追溯寫入暫存器陣列所追溯得到之該合併值以進行解碼；以及

一解碼暫存器陣列，根據已追溯得到之該合併值將儲存在該解碼暫存器陣列之該判斷位元進行解碼。

13.如申請專利範圍第 12 項所述之 Viterbi 解碼器的解碼電路，其中該解碼暫存器陣列所儲存之該判斷位元達到預定數量時，該解碼暫存器陣列變為該等待暫存器陣列。

14.如申請專利範圍第 13 項所述之 Viterbi 解碼器的解碼電路，其中當該追溯寫入暫存器陣列追溯得到該合併值並提供給該等待暫存器陣列，該等待暫存器陣列變為該解碼暫存器陣列，並根據該合併值進行解碼。

15.如申請專利範圍第 14 項所述之 Viterbi 解碼器的解碼電路，其中該追溯寫入暫存器陣列與該等待暫存器陣列之間更包括：

一第一狀態擷取電路，與該追溯寫入暫存器陣列進行雙向資料傳輸，並可送出資料；

一第一延遲暫存器，接收該第一狀態擷取電路所送出的資料，並延遲一段時間後輸出資料；以及

一第二狀態擷取電路，接收該第一延遲暫存器所送出的資料，並與該等待暫存器陣列進行雙向資料傳輸。

16.如申請專利範圍第 14 項所述之 Viterbi 解碼器的解碼電路，其中該追溯寫入暫存器陣列與該解碼暫存器陣列之間更包括：

## 六、申請專利範圍

一第三狀態擷取電路，與該追溯寫入暫存器陣列進行雙向資料傳輸，並可送出資料；

一第二延遲暫存器，接收該第三狀態擷取電路所送出的資料，並延遲一段時間後輸出資料；以及

一第四狀態擷取電路，接收該第二延遲暫存器所送出的資料，並與該解碼暫存器陣列進行雙向資料傳輸。

17.如申請專利範圍第 16 項所述之 Viterbi 解碼器的解碼電路，其中該第一狀態擷取電路與該暫存器陣列中之一記憶體區塊及一狀態暫存器做資料存取，該第一狀態擷取電路更包括：

一第一判斷電路，具有複數個判斷信號輸入端可接收該狀態暫存器所輸出的資料，並具有一選擇信號輸出端可輸出一選擇信號；

一第一多工器，具有複數個輸入端可接收該記憶體區塊的資料，具有一選擇端可接收該狀態暫存器的資料，並具有一輸出端可輸出資料；以及

一第二多工器，具有該些輸入端可接收該狀態暫存器與該第一多工器之該輸出端所輸出的資料，具有該選擇端可接收該第一判斷電路之該選擇信號輸出端所輸出之該選擇信號，並具有該輸出端可輸出資料至該狀態暫存器。

18.如申請專利範圍第 16 項所述之 Viterbi 解碼器的解碼電路，其中該第二狀態擷取電路與該暫存器陣列中之該記憶體區塊及該狀態暫存器進行資料存取，該第

(請先閱讀背面之注意事項再填寫本頁)

裝

訂

線

## 六、申請專利範圍

二狀態擷取電路更包括：

一第二判斷電路，具有該些判斷信號輸入端可接收該狀態暫存器所輸出的資料，並具有該選擇信號輸出端可輸出該選擇信號；

一第三多工器，具有該些輸入端可接收該記憶體區塊的資料，具有該選擇端可接收該狀態暫存器的資料，並具有該輸出端可輸出資料；以及

一第四多工器，具有該些輸入端可接收該狀態暫存器與該第三多工器之該輸出端所輸出的資料，具有該選擇端可接收該第二判斷電路之該選擇信號輸出端所輸出之該選擇信號，並具有該輸出端輸出資料至該狀態暫存器。

19.如申請專利範圍第 16 項所述之 Viterbi 解碼器的解碼電路，其中該第三狀態擷取電路與該暫存器陣列中之該記憶體區塊及該狀態暫存器進行資料存取，該第三狀態擷取電路更包括：

一第三判斷電路，具有該些判斷信號輸入端可接收該狀態暫存器所輸出的資料，並具有該選擇信號輸出端可輸出該選擇信號；

一第五多工器，具有該些輸入端可接收該記憶體區塊的資料，具有選擇端可接收該狀態暫存器的資料，並具有該輸出端可輸出資料；以及

一第六多工器，具有該些輸入端可接收該狀態暫存器與該第五多工器之該輸出端所輸出的資料，具有該選

## 六、申請專利範圍

擇端可接收該第三判斷電路之該選擇信號輸出端所輸出之該選擇信號，並具有該輸出端輸出資料至該狀態暫存器。

20.如申請專利範圍第 16 項所述之 Viterbi 解碼器的解碼電路，其中該第四狀態擷取電路與該暫存器陣列中之該記憶體區塊及該狀態暫存器進行資料存取，該第四狀態擷取電路更包括：

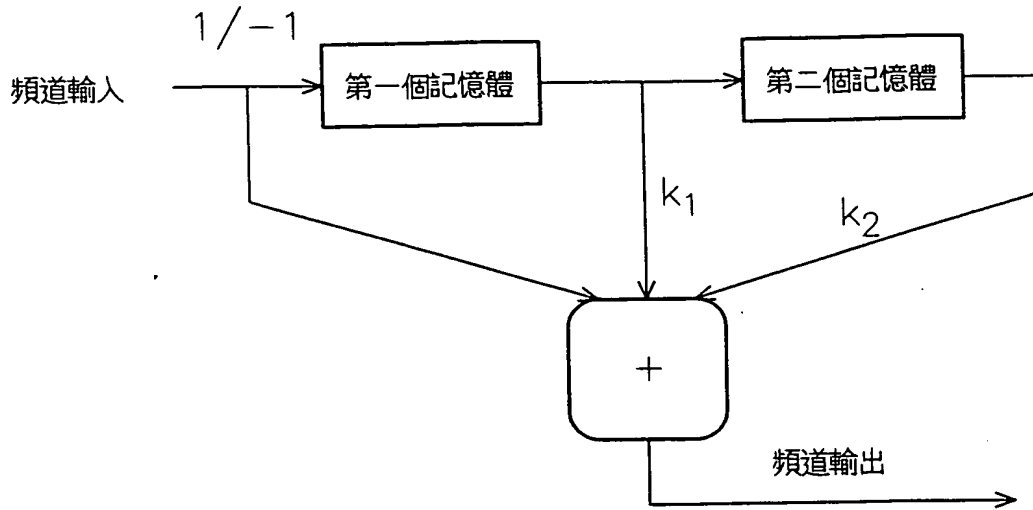
一第四判斷電路，具有該些判斷信號輸入端可接收該狀態暫存器所輸出的資料，並具有該選擇信號輸出端可輸出該選擇信號；

一第七多工器，具有該些輸入端可接收該記憶體區塊的資料，具有該選擇端可接收該狀態暫存器的資料，並具有該輸出端可輸出資料；以及

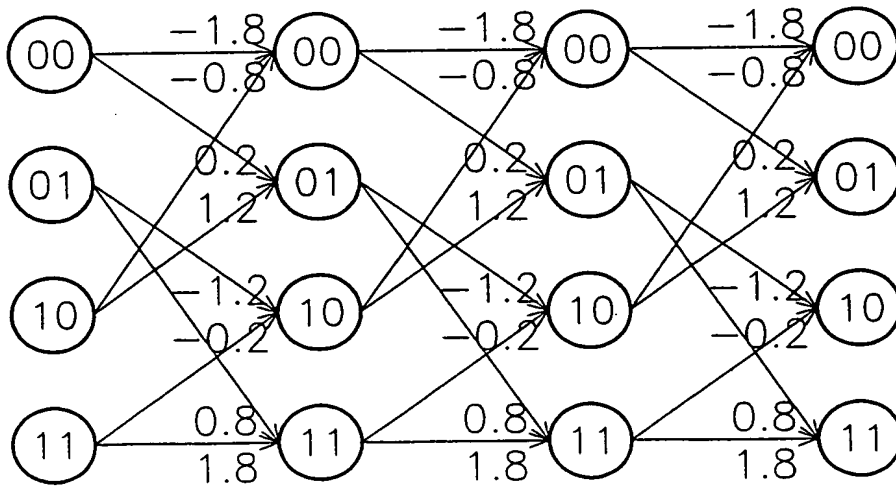
一第八多工器，具有該些輸入端可接收該狀態暫存器與該第七多工器之該輸出端所輸出的資料，具有該選擇端可接收該第四判斷電路之該選擇信號輸出端所輸出之該選擇信號，並具有該輸出端可輸出資料至該狀態暫存器。

(請先閱讀背面之注意事項再填寫本頁)

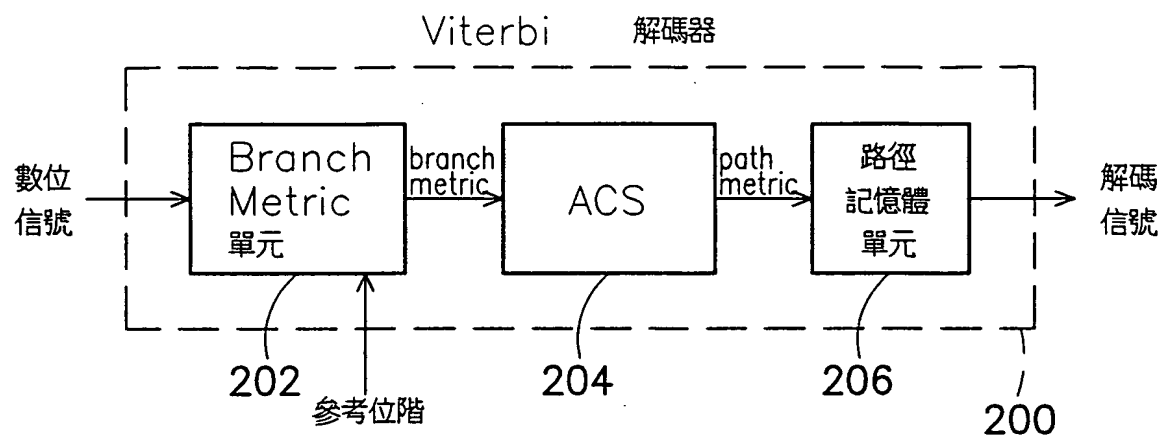
裝  
訂  
線



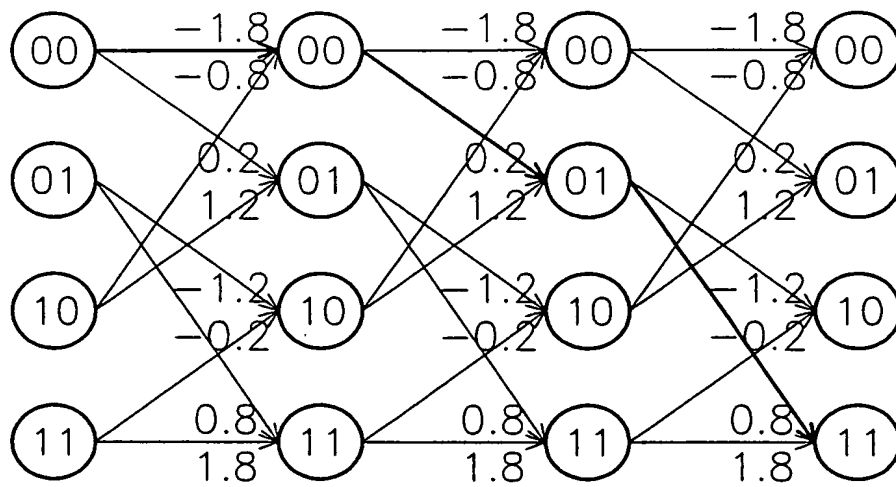
第 1A 圖



第 1B 圖



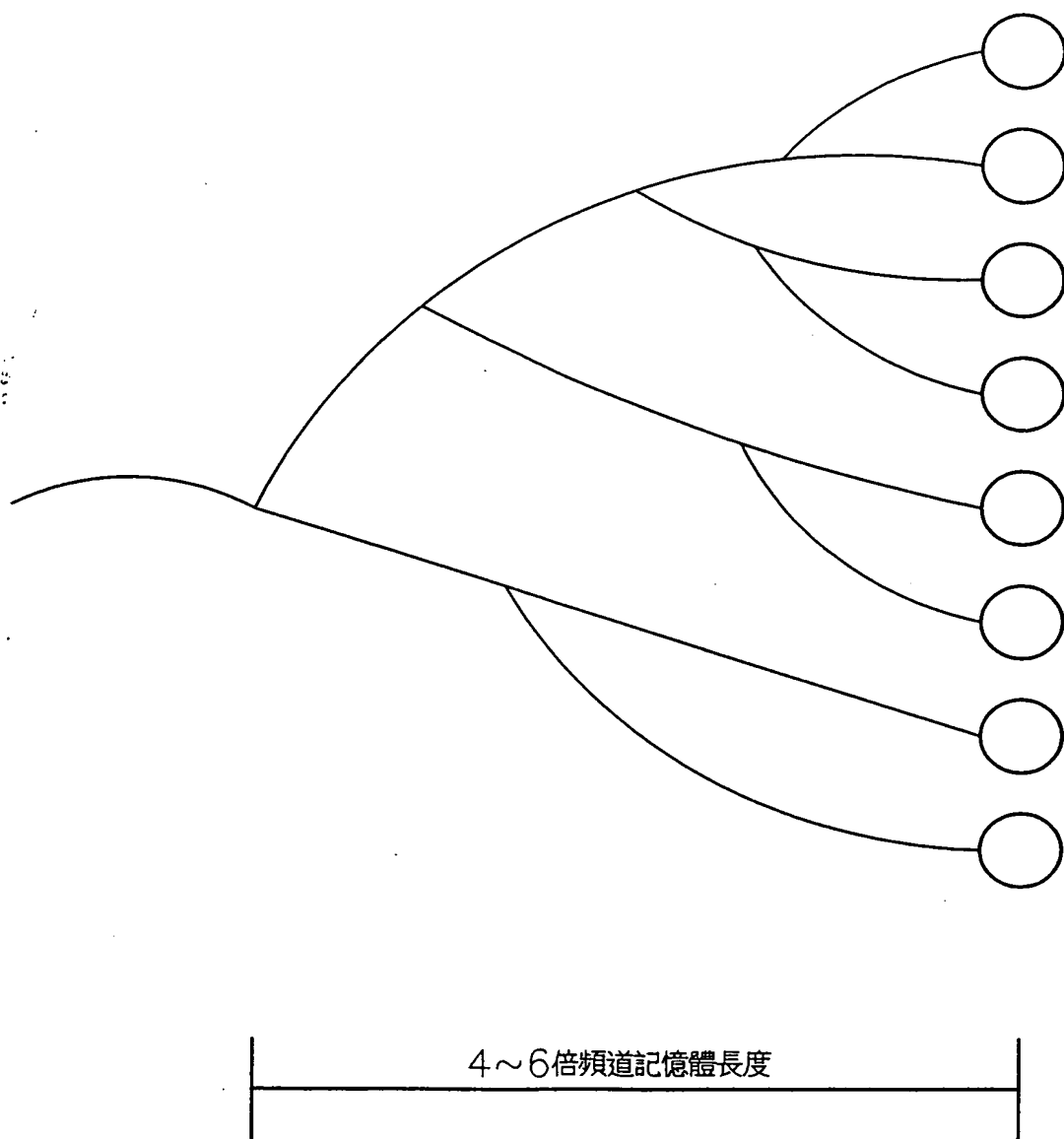
第 2 圖



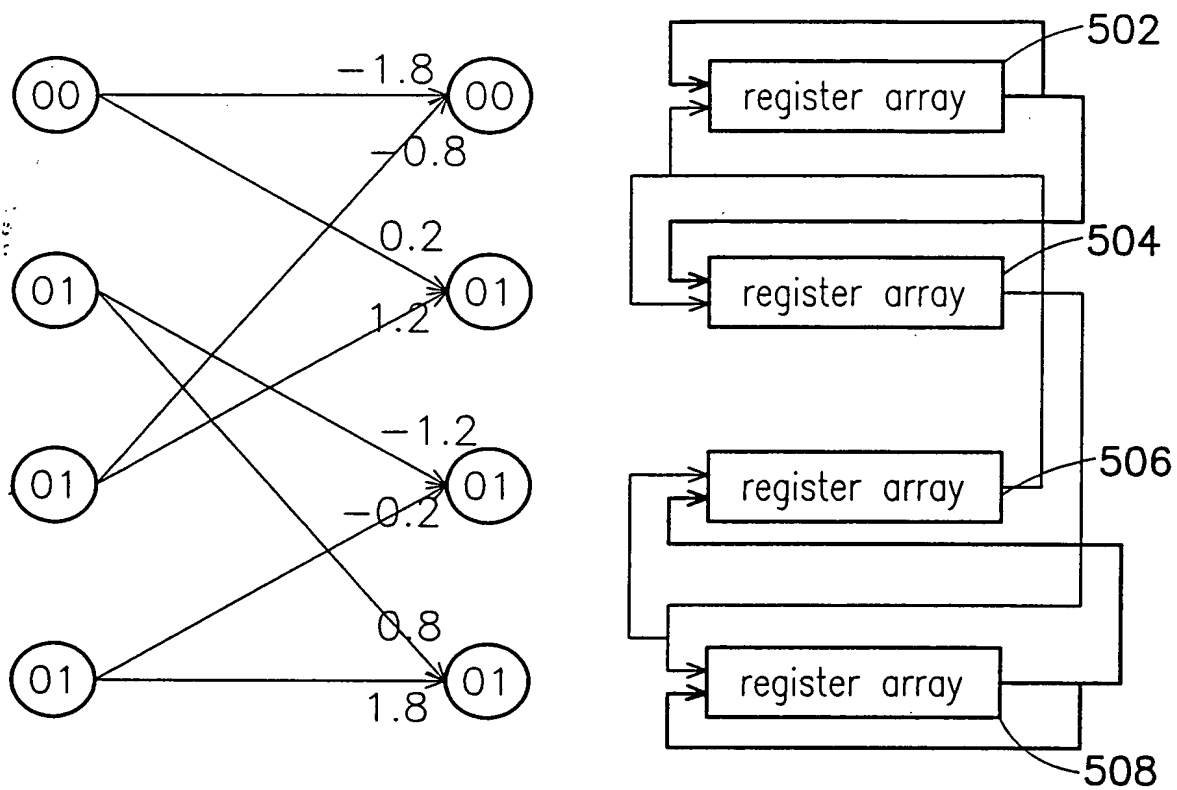
| 接收信號R1 | 接收信號R2 | 接收信號R3 |     |
|--------|--------|--------|-----|
| 在1T    | 在2T    | 在3T    | 在4T |

第 3 圖

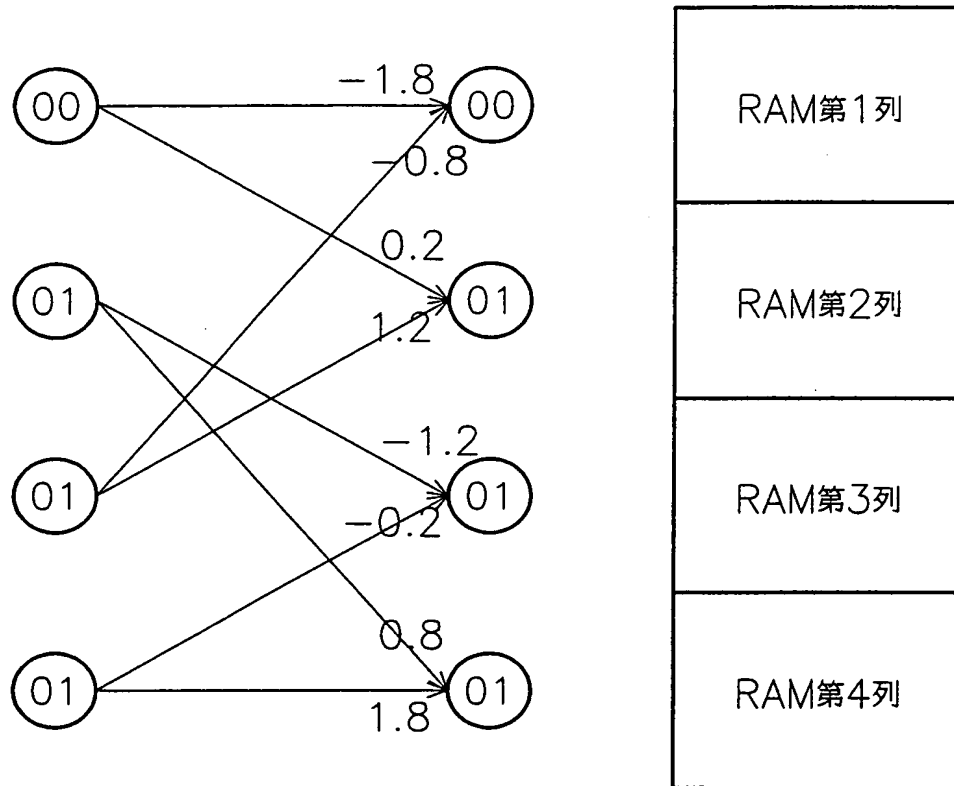




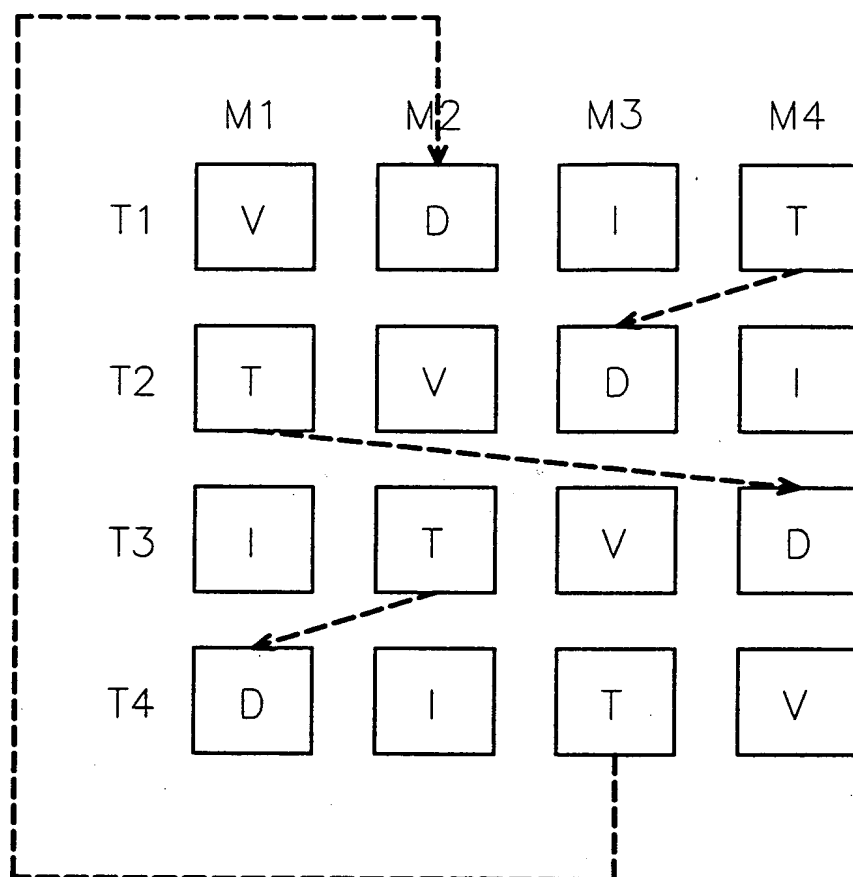
第 4 圖



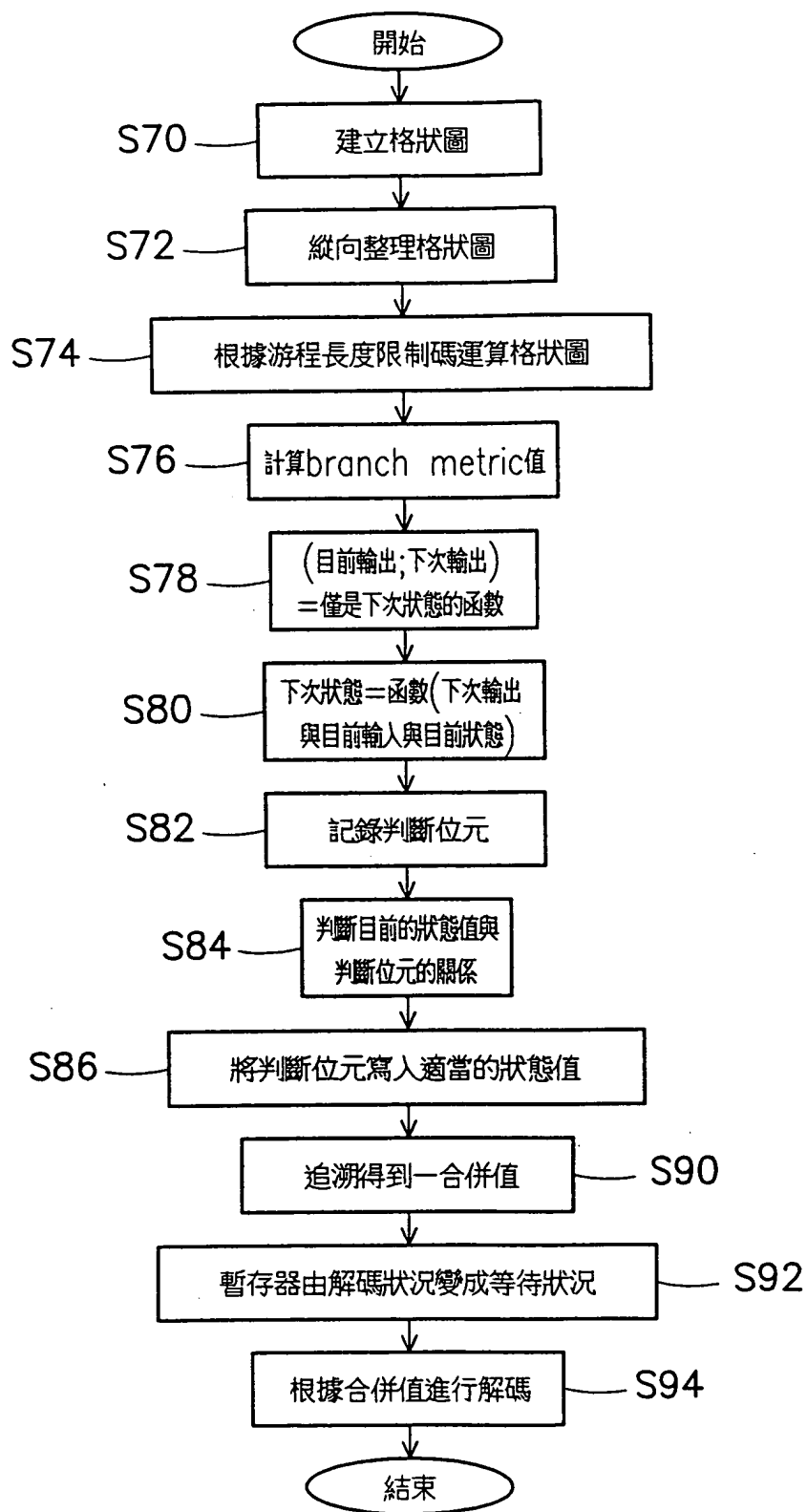
第 5 圖



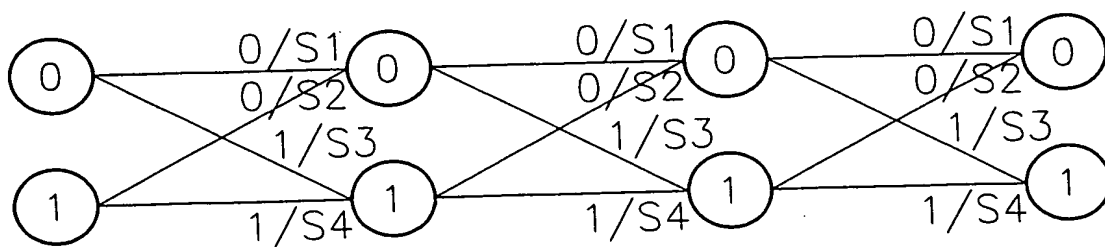
第 6 圖



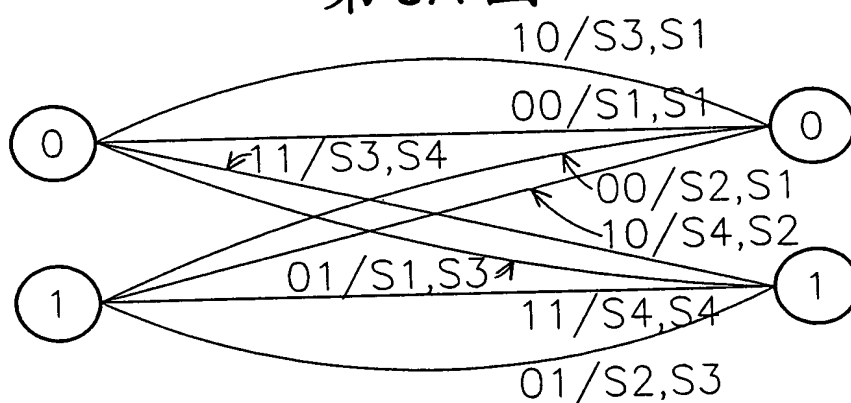
第 7 圖



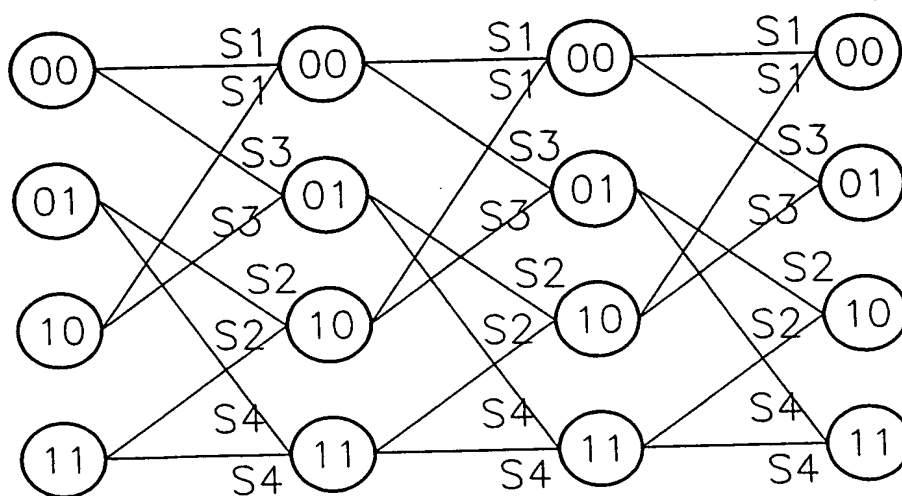
第 8 圖



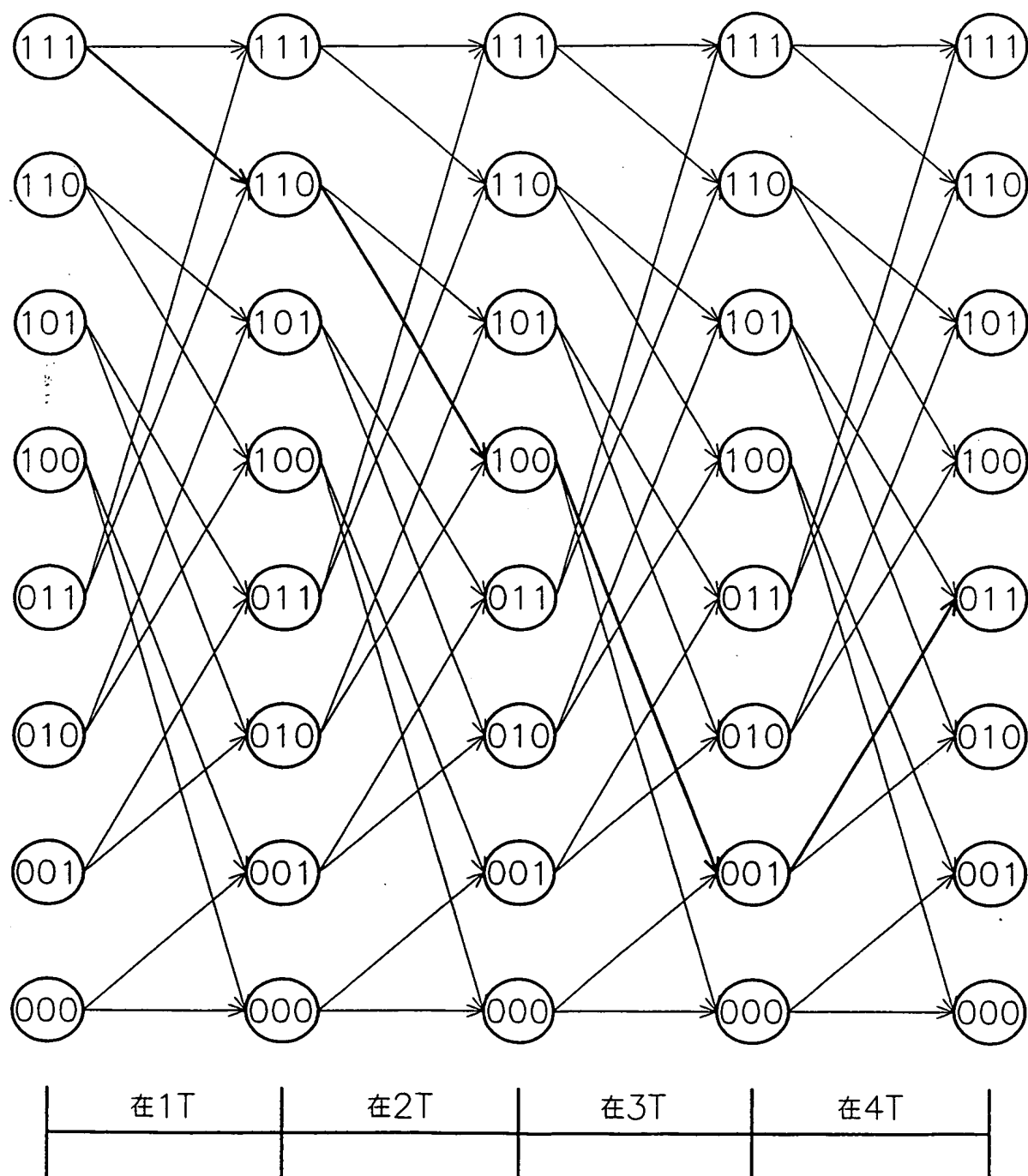
第9A圖



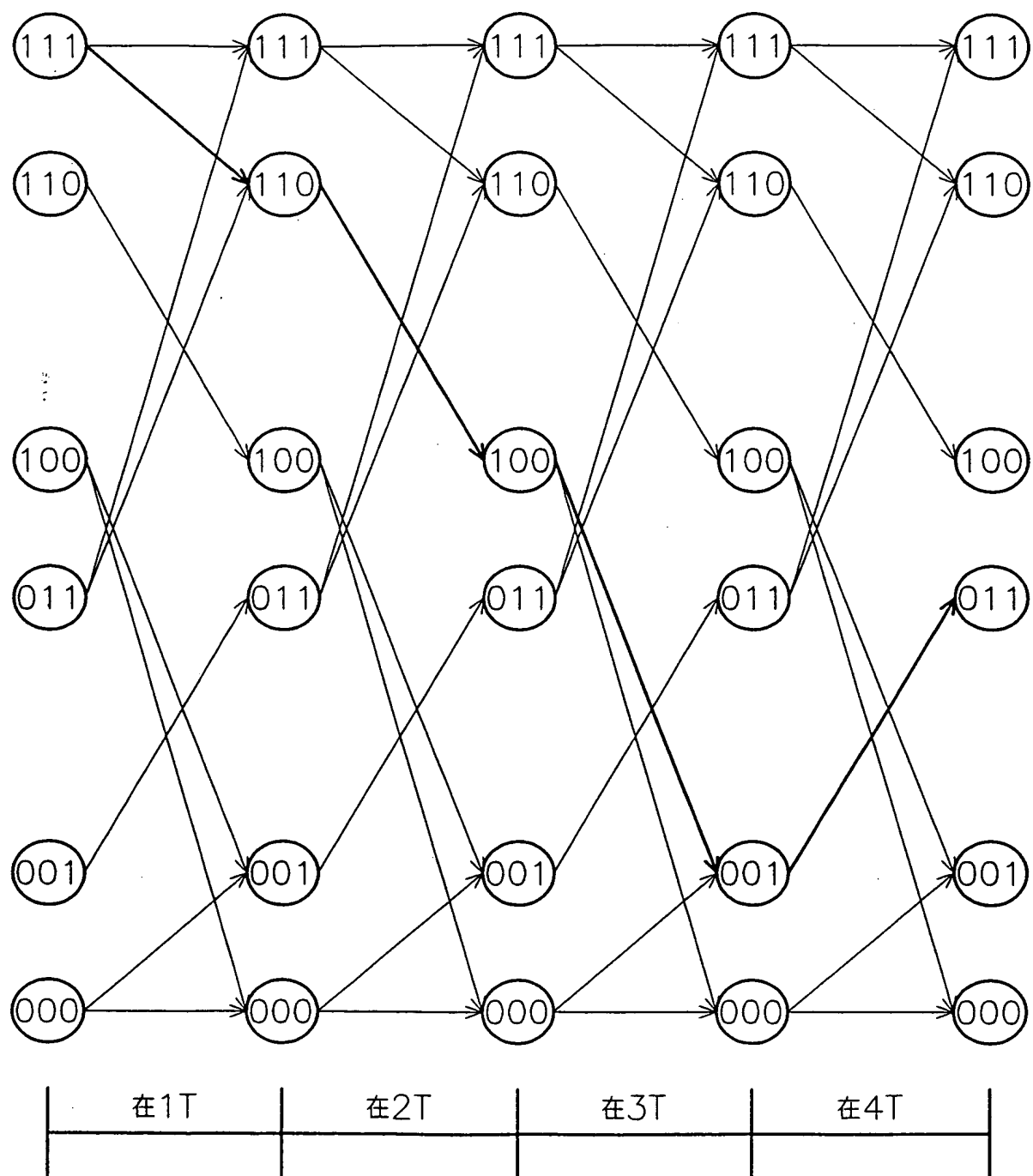
第9B圖



第9C圖

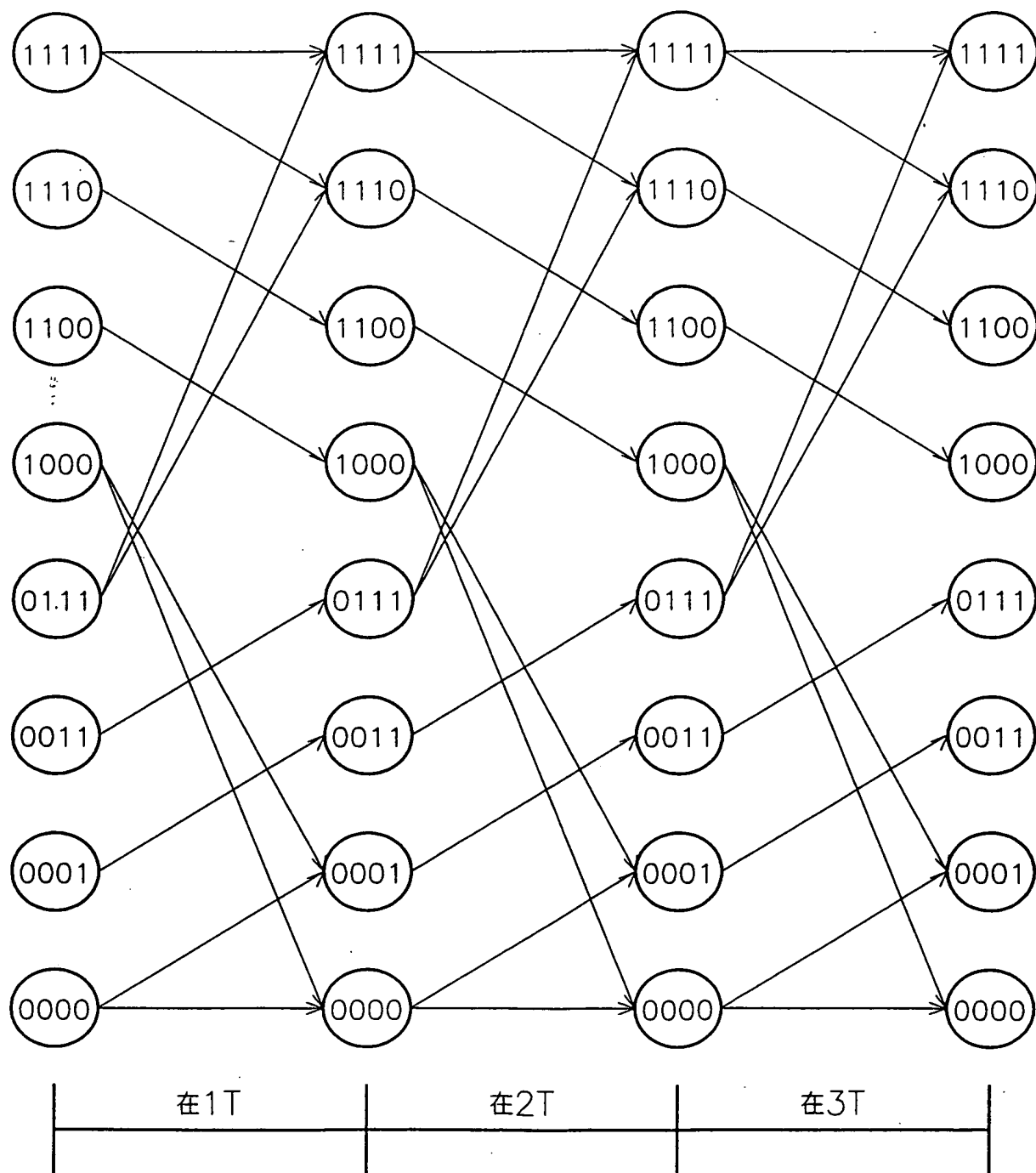


第 10 圖



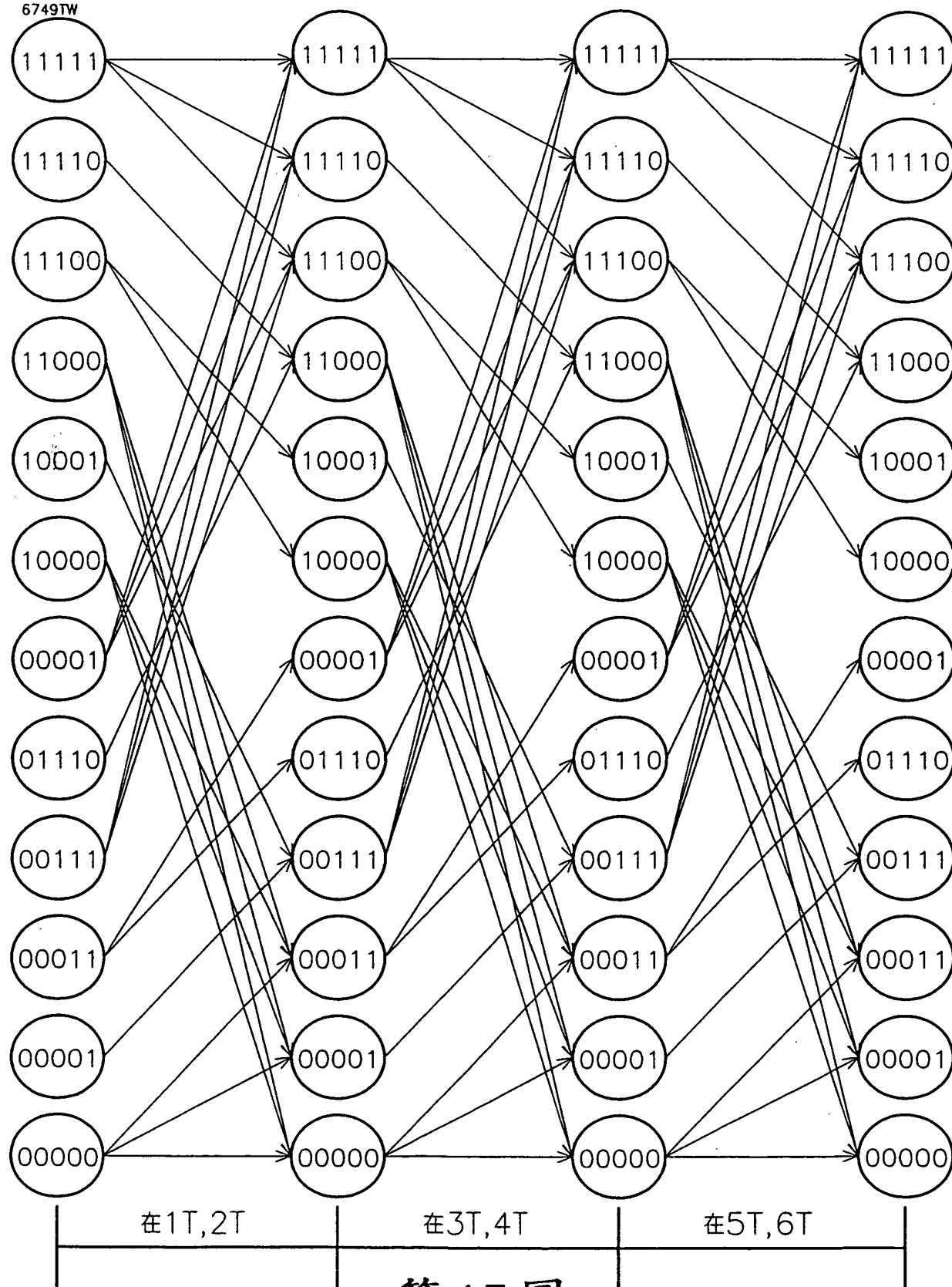
第 11 圖



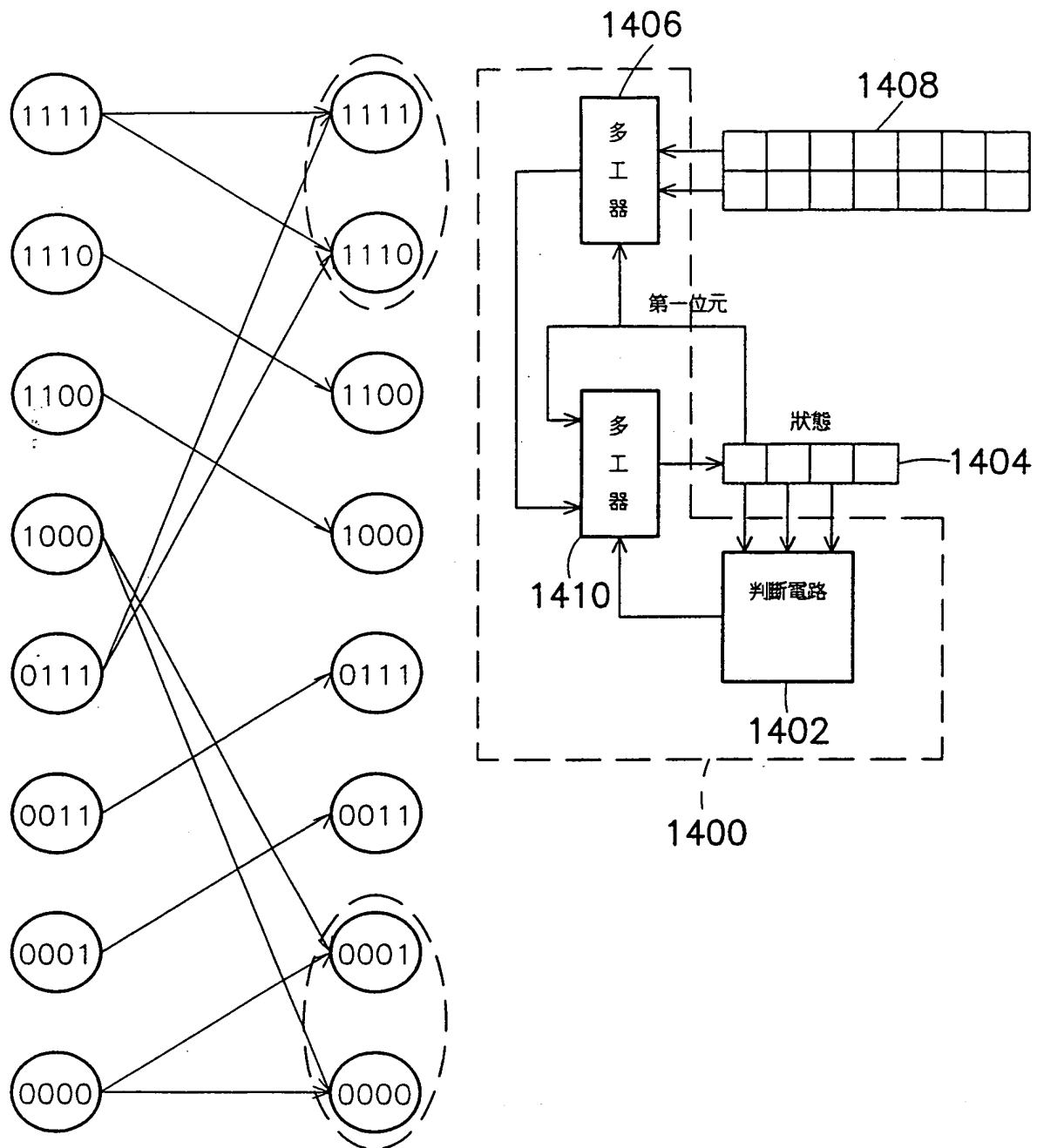


第 12 圖

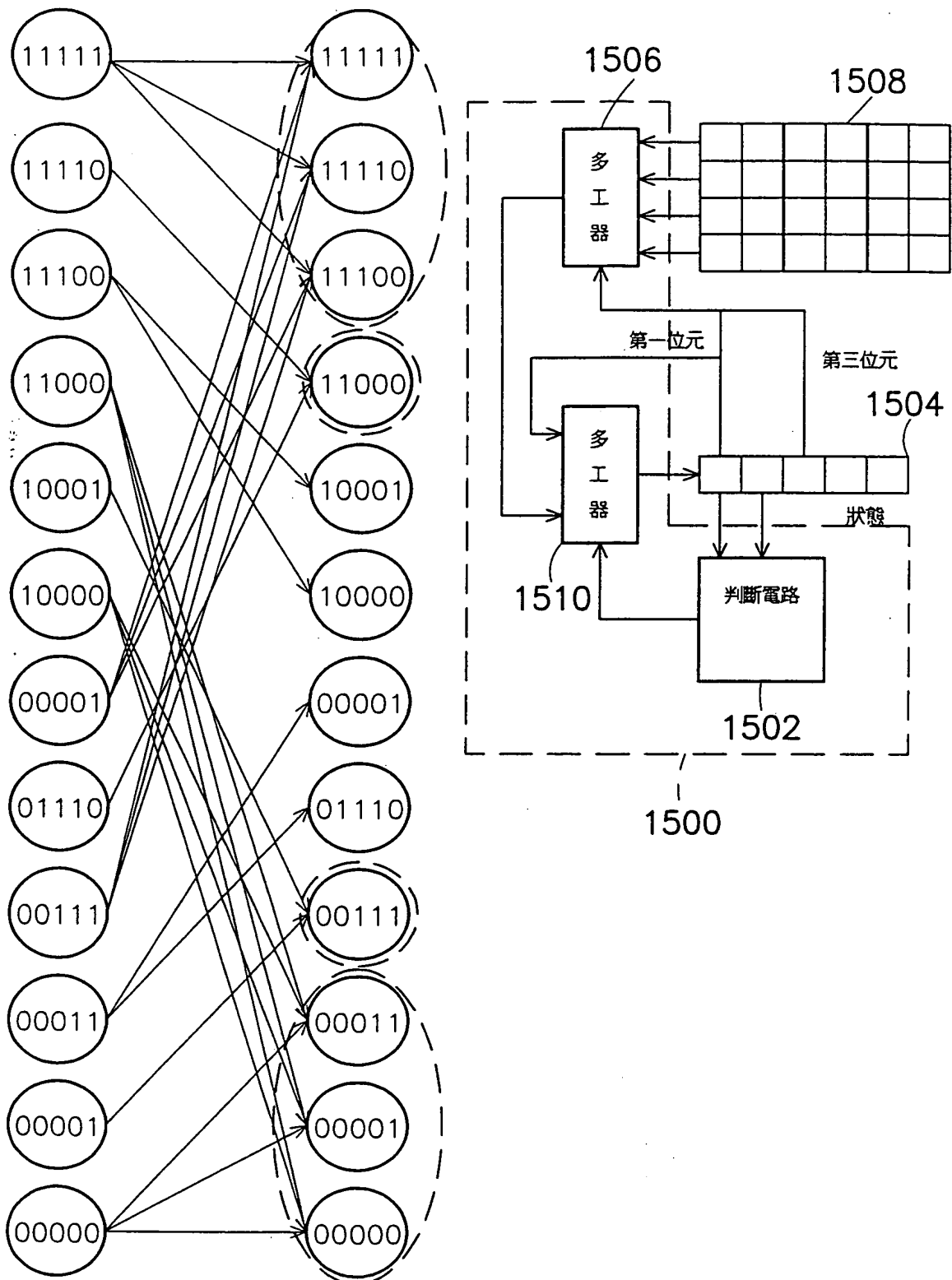
6749TW



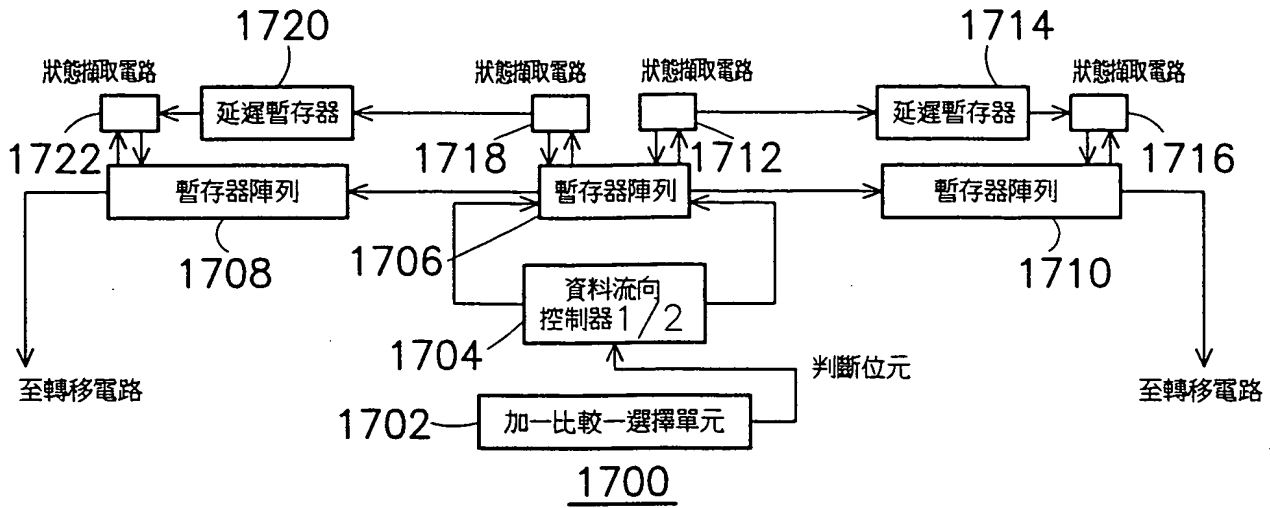
第 13 圖



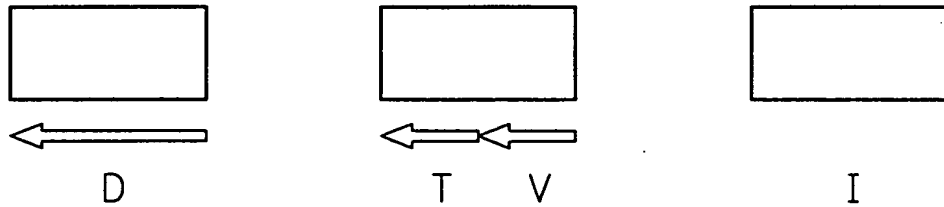
第 14 圖



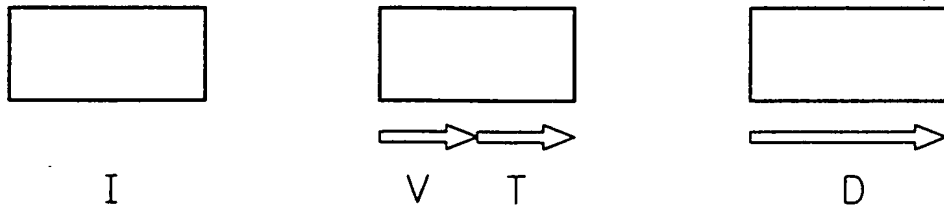
第 15 圖



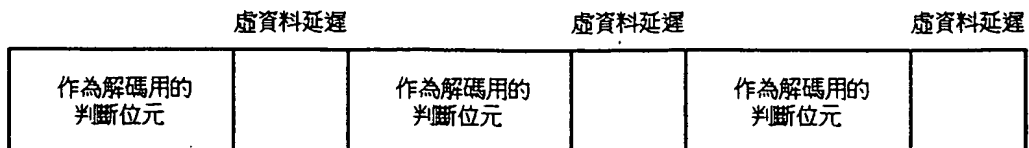
第17A圖



第17B圖



第17C圖



第17D圖